SANDA CHERATA, TEODOR VUŞCAN, EMMA TĂMÂIANU

SILEX – FUNCŢIILE DE LEMATIZARE ŞI DE GENERARE A PARADIGMELOR

Consideraţii introductive

Cele de faţă reprezintă al doilea articol[1] din seria dedicată prezentării sistemului lexico-morfologic computerizat pentru analiza textelor româneşti (**SILEX**), sistem realizat în cadrul unei colaborări ample între un colectiv de la S.C. Software ITC S.A. şi Centrul de Analiză a Textului de la Facultatea de Litere a Universităţii „Babeş-Bolyai".

> *Precizări terminologice. formă* = unitate grafică ce corespunde unităţii lingvistice 'cuvânt-ocurenţă'; *lemă* = unitate lexicală, reprezentată în mod convenţional printr-o formă de bază (exemplu infinitivul prezent activ al verbului, nominativul singular nearticulat al substantivului etc.) şi apartenenţa ei la o anumită clasă lexico-gramaticală; *rădăcină* = şirul de caractere invariant în cursul flexiunii (pentru întreaga paradigmă sau doar pentru o parte a acesteia); *Sinv* = segmentul de cuvânt comun (invariant al) unei părţi a paradigmei, prezent sub formă de câmp al intrării din dicţionar corespunzătoare unei leme; *terminaţie* = şirul de caractere ce se adaugă 'rădăcinii' pentru a obţine o formă flexionată a cuvântului.

Dicţionarul SILEX a fost descris pe larg (intrări de bază, atribute, structurare internă, optimizări, aplicaţii) în primul articol; cel de faţă se focalizează asupra componentei dinamice, adică asupra modulelor de lematizare şi de generare a paradigmelor. În conceperea şi realizarea celor două module, autorii au încorporat sugestii din bibliografia de specialitate (vezi **Sursele bibliografice**), însă produsul îşi păstrează un marcat caracter de originalitate în spaţiul prelucrării automate a textelor româneşti, nu doar sub aspectul său strict informatic, ci şi în ce priveşte soluţiile de formalizare a descrierii lingvistice.

1. Procesul de lematizare

Lematizarea reprezintă procesul prin care fiecare formă ocurentă într-un text este încadrată lemei sale. Sistemul SILEX este astfel conceput încât permite,

---

[1] Vezi *SILEX – Un sistem lexico-morfologic computerizat pentru analiza textelor româneşti*, în „Dacoromania", serie nouă, I, 1994-1995, nr. 1-2.

DACOROMANIA serie nouă, II, 1996-1997, Cluj-Napoca, p. 273-285

pentru fiecare lemă, determinarea tuturor formelor ce aparţin paradigmei sale, prin intermediul unui mecanism care nu presupune înregistrarea fizică a formelor. Lematizarea formelor este procesul prin care se realizează funcţia inversă: fiind dată o formă, se determină lema (lemele) în a cărei (căror) paradigmă apare forma în cauză; concomitent cu determinarea acestor leme, se regăsesc şi atributele aferente formei.

Observaţie. Convenim că paradigma unui cuvânt neflexibil constă dintr-o singură formă, coincidentă cu lema.

Procesul de lematizare va furniza, pentru fiecare formă, următoarele rezultate:

– lema, adică reprezentantul paradigmei cuvântului;
– clasa lexico-gramaticală a cuvântului (substantiv, adjectiv, verb etc.);
– subclasa lexico-gramaticală, dacă este cazul (substantiv propriu sau comun, verb predicativ, auxiliar sau copulativ etc.);
– în cazul cuvintelor flexibile, se stabilesc în plus celelalte determinări, dependente de clasa morfologică:
  – pentru substantive şi adjective: *gen, număr, caz*;
  – pentru verbe: *mod, timp, persoană, număr*;
  – pentru pronume: *felul pronumelui, gen, număr, caz, persoană*;
  – pentru numerale: *felul numeralului, gen, caz* (unde există).

Precizare. În versiunea actuală, analiza formelor se face strict acontextual, furnizându-se, pentru fiecare ocurenţă a unei forme, toate încadrările morfologice posibile, adică toate lemele în paradigma cărora apare forma respectivă, împreună cu atributele aferente formei flexionate. De exemplu, pentru forma ***pară***, rezultatele procesului de lematizare sunt următoarele:

– *pară*, substantiv, feminin, singular, nominativ/acuzativ;
– *par*, adjectiv, feminin, singular, nominativ/acuzativ;
– *a părea*, verb, conjunctiv, prezent, persoana a III-a, singular;
– *a para*, verb, indicativ, perfect simplu, persoana a III-a, singular.

Observaţie. Pentru situaţiile de omonimie gramaticală în interiorul aceleiaşi paradigme se afişează numai una dintre valorile categoriilor implicate în omonimie, şi anume cea care apare prima în ordonarea uzuală din lucrările de morfologie. (Vezi, de exemplu, în **Anexa 1**: pentru forma verbală *pustii*, se afişează valoarea 'infinitiv', nu şi valoarea 'indicativ, perfect simplu'.)

1. 1. Metoda utilizată în procesul de lematizare

Textul de prelucrat poate fi introdus fie de la terminal, fie dintr-un fişier. Se preia o linie a textului sursă şi se lematizează pe rând fiecare cuvânt.

În procesul de lematizare, pentru fiecare cuvânt de prelucrat se fac toate descompunerile posibile în două părţi:

$$cuvânt = \mathbf{r} + \mathbf{t}$$

unde **r** corespunde unei eventuale rădăcini (parte invariantă în cadrul paradigmei), iar **t** corespunde unei eventuale terminații a cuvântului. În descompunerea inițială, **t** este șirul vid.

Pentru fiecare descompunere de acest fel se verifică dacă poate fi găsită o lemă care să-i corespundă. În acest scop, se caută în dicționarul SILEX dacă există o intrare al cărei câmp *Sinv* coincide cu **r**. În cazul în care o astfel de intrare este găsită, se verifică dacă **t** corespunde unei terminații din lista de terminații (*lst_ter*) asociată intrării respective din dicționar. În caz afirmativ, se determină lema și atributele aferente terminației.

Observație. În principiu, în dicționar pot exista mai multe intrări care să aibă câmpul *Sinv* coincident cu **r** (din cauză că același segment invariant poate apărea în paradigma mai multor leme). De aceea, pentru un **r** dat se analizează toate intrările respective. Pentru fiecare intrare găsită se face verificarea terminației **t**, detectându-se astfel toate lemele în a căror paradigmă apare cuvântul dat, *în descompunerea tratată.*

În continuare se încearcă o altă descompunere a cuvântului, de forma:

cuvânt = **r'** + **t'**,

unde:

r = **r'** + *u*

t' = *u* + **t**,

u fiind ultima literă a lui **r**, iar **r** și **t** – elementele descompunerii imediat anterioare, și se repetă aceleași operații de verificare. Este posibil ca mai multe descompuneri să fie valide (în cazul formelor omografe). În felul acesta se determină **toate** încadrările posibile (se identifică toate lemele în a căror paradigmă apare forma în cauză). Procesul de descompunere se încheie în momentul în care fie r' este șirul vid, fie t' nu aparține listelor de terminații.

În vederea îmbunătățirii performanțelor de timp ale metodei de lematizare prezentate mai sus, s-au făcut o serie de optimizări atât (a) în ce privește algoritmul propriu-zis, cât și (b) în ce privește conținutul dicționarului. Optimizările (a), cele privind procesul de descompunere, reduc numărul de descompuneri posibile ale unui cuvânt dat, iar optimizările (b), cele de la nivelul dicționarului, reduc numărul de rădăcini omografe.

1. 2. Implementarea procesului de lematizare

1. 2. 1. Etape:

(i) se stabilesc opțiunile privitoare la modalitatea de introducere a textului de analizat, prin dialog cu utilizatorul;

(ii) se inițializează datele globale și se deschid bazele de date și fișierele utilizate;

(iii) se preia o linie de text şi se separă cuvintele din linie;
(iv) se analizează pe rând cuvintele din linia de text;
(v) se înscriu rezultatele în fişierele de rezultate.

1. 2. 2. Descriere

În cele ce urmează prezentăm detaliat cele mai importante dintre aceste etape.

La lansarea în execuţie a programului de recunoaştere, apare o fereastră de dialog în care utilizatorul stabileşte modalitatea de introducere a textului, şi anume: de la tastatură sau dintr-un fişier.

În cazul în care textul este introdus dintr-un fişier, rezultatul analizei va fi memorat într-un fişier având acelaşi nume ca fişierul sursă şi extensia „*.rec*". Cuvintele nerecunoscute în urma analizei vor fi memorate într-o bază de date având acelaşi nume ca fişierul sursă şi extensia „*.nec*". Menţionăm că această facilitate este utilă pentru punerea la punct a sistemului de recunoaştere a formelor.

În continuare se citeşte câte o linie a textului sursă şi se prelucrează cuvintele liniei unul câte unul. Pentru aceasta, se separă cuvântul analizat ţinând cont de semnele de punctuaţie şi de spaţiile separatoare. În acest moment se detectează şi dacă forma este precedată sau urmată de cratimă. Cuvântul astfel delimitat este convertit în litere mici, deoarece în dicţionar intrările conţin numai astfel de litere. La conversie se ţine cont şi de codurile speciale pentru diacritice.

Se trece apoi la recunoaşterea cuvântului în cauză, conform metodei prezentate sub **1. 1.** Pentru aceasta, se încearcă întâi recunoaşterea cuvântului aşa cum este dat. Dacă recunoaşterea nu reuşeşte se mai fac următoarele încercări:

a. Se testează dacă forma începe cu unul din prefixele *re-*, *ne-*, *nemai-*, prefixe foarte productive mai ales pentru rădăcinile verbale[2]. În caz afirmativ, se încearcă recunoaşterea unui verb sau derivat verbal pornind de la forma dată, din care s-a eliminat prefixul. Dacă recunoaşterea a reuşit, se construieşte lema corespunzătoare cuvântului prin adăugarea prefixului la lema găsită în dicţionar.

b. Când forma tot nu este recunoscută, se testează dacă este precedată sau urmată de cratimă, caz în care aceasta poate semnifica absenţa unei litere. În caz afirmativ, se încearcă adăugarea unei litere, conform situaţiilor de folosire a cratimei specificate în gramatica limbii române şi se încearcă din nou recunoaşterea.

În **Anexa 1** prezentăm un exemplu cu rezultatele analizei unui text real (poemul *Amurg* de G. Bacovia).

---

[2] Din motive de optimizare a căutărilor în dicţionar, pentru respectivele derivate nu există intrări proprii. Vezi, în acest sens, precedentul nostru articol, sub **2. 2. 1.**

Precizări la **1. 2. 2.** (1) Dat fiind numărul relativ mare, în raport cu ansamblul dicţionarului, al lemelor din categoriile: infinitiv lung, adjectiv participial şi adjectiv derivat din verb prin sufixul *-tor*, s-au făcut optimizări de spaţiu şi timp prin tratarea specială a paradigmelor acestor cuvinte, în absenţa intrărilor proprii în dicţionar. (2) Tot din motive de optimizare a spaţiului şi timpului, în versiunea actuală dicţionarul propriu-zis este împărţit în două baze de date, după lungimea cuvintelor.

Au fost consecvent respectate cerinţele de modularizare şi de structurare flexibilă, încât acestea să facă posibile extensii ulterioare.

1. 3. Concluzii privind procesul de lematizare

Funcţia de lematizare este implementată într-un modul separat (independent) al sistemului SILEX, pentru a asigura astfel posibilitatea de a dezvolta pe baza ei diverse aplicaţii de mare actualitate. De pildă, în prezent se află într-o fază avansată de realizare o aplicaţie de verificare automată a textelor româneşti în privinţa greşelilor de ortografie, aplicaţie deosebit de utilă în activitatea editurilor. În aceeaşi ordine de idei, la Centrul de Analiză a Textului de la Facultatea de Litere a Universităţii „Babeş-Bolyai" se află în curs de experimentare un sistem pentru concordanţele textelor poetice româneşti, sistem care încorporează acest modul de lematizare.

2. Procesul de generare a paradigmei cuvintelor flexibile

Prin 'proces de generare a paradigmei unei leme date' se înţelege construcţia formelor unei paradigme pornind de la reprezentantul ei (**lema**) şi de la clasa lexico-gramaticală. Din cauza omografiilor, aceeaşi lemă poate avea paradigme diferite, aparţinând aceleiaşi clase morfologice sau unor clase morfologice diferite. Distincţia dintre două leme omografe care corespund la clase morfologice diferite se face prin atributul de clasă lexico-gramaticală ataşat lemei şi existent în dicţionar. Distincţia dintre două leme omografe care corespund aceleiaşi clase morfologice se face prin intermediul listei de terminaţii, ataşată în dicţionar lemei în cauză, iar în cazul substantivelor distincţia se face ţinând cont şi de atributul gen. Pentru lemele omografe având aceeaşi clasă lexico-gramaticală se vor genera toate paradigmele.

2. 1. Metoda utilizată în procesul de generare a paradigmelor

Flexionarea este realizată ca o operaţie simetrică faţă de lematizare, atât în privinţa metodei, cât şi în cea a implementării.

Procesul de flexionare utilizează următoarele informaţii existente în bazele de date ale SILEX:

(i) dicţionarul SILEX, ordonat de data aceasta după leme, permite asocierea rapidă şi simplă a tuturor paradigmelor corespunzătoare unei leme date şi unei clase lexico-gramaticale date; separarea paradigmelor corespunzătoare aceleiaşi leme (cazul omografiilor) se face prin intermediul atributului 'listă de terminaţii' ataşat unei paradigme a lemei;

(ii) listele de terminaţii structurate pe clase lexico-gramaticale şi, în cazul substantivului, pe genuri.

Pentru flexionarea unui cuvânt este necesară, aşadar, precizarea lemei şi a clasei ei lexico-gramaticale.

2. 1. 1. Etape

Procesul de flexionare a unui cuvânt se realizează în următoarele etape:

(1) se caută intrarea de dicţionar corespunzătoare lemei şi clasei date;

(2) se reţine câmpul *Sinv* (rădăcina), numărul listei de terminaţii şi subparadigma corespunzătoare rădăcinii;

(3) ţinând cont de subparadigma asociată rădăcinii, se construieşte, pentru numărul curent al listei de terminaţii, partea corespunzătoare din paradigma cuvântului, prin ataşarea terminaţiilor la rădăcină;

(4) dacă paradigma nu este generată complet, se caută în continuare o altă intrare din dicţionar corespunzătoare lemei, clasei şi listei de terminaţii curente şi se procedează ca mai sus; se repetă pasul 4 până la obţinerea paradigmei complete;

Obs. La substantivele şi verbele defective, respectivul fenomen este tratat atât la nivelul dicţionarului, prin informaţii adecvate, cât şi în procesul de flexionare, pentru determinarea exactă a paradigmei.

(5) în eventualitatea omografiilor din cadrul aceleiaşi clase morfologice se reia procedeul de la punctul (1); procesul de generare a paradigmelor se încheie după ce s-au tratat toate intrările din dicţionar corespunzătoare lemei şi clasei lexico-gramaticale specificate.

2. 2. Implementarea procesului de flexionare

Etape:

(i) se iniţializează datele globale şi se deschid bazele de date şi fişierele utilizate; pentru buna funcţionare a procesului de flexionare, informaţiile din dicţionar sunt ordonate după câmpul **lemă;**

(ii) se citesc lema şi clasa introduse de utilizator;

(iii) se generează o paradigmă corespunzătoare lemei date şi se afişează în formatul extern care apare în lucrările de morfologie;

(iv) în cazul omografiilor, se generează pe rând şi se afişează toate paradigmele corespunzătoare lemei şi clasei date.

Formatul de prezentare a paradigmei este dependent de clasa lexico-gramaticală a cuvântului flexionat. În **Anexa 2** prezentăm un exemplu cu rezultatele flexionării mai multor cuvinte.

2. 3. Concluzii privind procesul de flexionare

Modulul de flexionare este astfel conceput încât să poată fi încorporat cu uşurinţă în diverse aplicaţii.

O primă utilizare reală a lui a fost făcută la verificarea corectitudinii datelor din dicționar și din listele de terminații.

Se preconizează ca modulul să fie încorporat în aplicațiile care folosesc dicționarul computerizat, pentru a oferi utilizatorului un mijloc de verificare a datelor introduse ulterior spre a îmbogăți dicționarul furnizat. În plus, modulul de flexionare poate fi încorporat în aplicații de natură didactică, de învățare asistată de calculator a morfologiei limbii române.

3. Considerații finale

Tratarea computerizată a textelor românești nu se poate realiza în absența unui instrument cum este **SILEX**. Modulele de lematizare și flexionare fac posibilă dezvoltarea diverselor aplicații vizând prelucrarea automată a textelor românești. Enumerăm doar câteva dintre aceste aplicații, în ordinea crescândă a complexității lor funcționale:

a. dicționar ortografic și morfologic computerizat al limbii române, ușor de întreținut și îmbogățit, furnizabil atât în formă computerizată, cât și în formă tipărită;

b. corector ortografic și morfologic pentru textele românești;

c. sistem pentru studii de statistică lexicală și gramaticală asupra textelor literare;

d. sistem de realizare a concordanțelor pentru operele literare românești, cu lematizare în mare parte automată;

e. suport didactic pentru studierea asistată de calculator a gramaticii limbii române în învățământul preuniversitar și pentru învățarea limbii române ca limbă străină (ortografie, morfologie și lexic).

ANEXA 1

Exemplu de analiză a unui text dat

AMURG
Trec corbii – ah, „Corbii"
Poetului Tradem –
Și curg pe-nnoptat
Pe-un frig înghețat

Se duc pe pustii...
Pe când, de argint,
În amurg de argint,
S-aprinde crai-nou,

Pe zări argintii
În vastul cavou...
Iubito ah, „Corbii"
Poetului Tradem...

Forma	Lema	Atribute
amurg	* amurg	* sb * n,N/A sg.neart.
trec	* trece	* vb * Indicativ,I,sg.
corbii	* corb	* sb * m,N/A pl. art.
ah	* ah	* in *
corbii	* corb	* sb * m,N/A pl. art.
poetului	* poet	* sb * m,G/D sg. art.
şi	* şi	* av *
şi	* şi	* cj *
şi	* sine	* pn *refl,mf,III,sg.,D
curg	* curge	* vb * Indicativ,I,sg.
pe	* pe	* pp *
nnoptat	* înnopta	* vb * Participiu
nnoptat	* înnoptat	* sb * m,N/A sg.neart.
pe	* pe	* pp *
un	* un	* ar * m,N/A sg.
un	* un_1	* aj * nehot,m,N/A sg.
frig	* frige	* vb * Indicativ,I,sg.
frig	* frig	* sb * n,N/A sg. neart.
îngheţat	* îngheţa	* vb * Participiu
îngheţat	* îngheţat	* aj * m,N/A sg. neart.
se	* sine	* pn * refl,mf,III,sg.,A
duc	* duce	* vb * Indicativ,I,sg.
pe	* pe	* pp *
pustii	* pustii	* vb * Infinitiv
pustii	* pustiu	* aj * m,N/A,pl.,neart.
pe	* pe	* pp *
când	* când	* av *
de	* de	* cj *
de	* de	* pp *
argint	* argint	* sb * m,N/A sg. neart.
s	* sine	* pn * refl,mf,III,sg.,A
aprinde	* aprinde	* vb * Indicativ,III,sg.

crai	* crai	* sb * m,N/A sg. neart.
nou	* nou	* aj * m,N/A sg. neart.
pe	* pe	* pp *
zări	* zări	* vb * Infinitiv
zări	* zare	* sb * f,N/A sg. neart.
argintii	* argintiu	* aj * m,N/A pl. neart.
în	* în	* pp *
vastul	* vast	* aj * m,N/A sg. art.
cavou	* cavou	* sb * n,N/A sg. neart.
ah	* ah	* in *
corbii	* corb	* sb * m,N/A pl. art.
poetului	* poet	* sb * m,G/D sg. art.

ANEXA 2

Exemple de flexionare a unor leme date

cască – substantiv feminin

N. o	cască	nişte	căşti	casca	căştile
D. unei	căşti	unor	căşti	căştii	căştilor
G. unei	căşti	unor	căşti	căştii	căştilor
A. o	cască	nişte	căşti	casca	căştile

copil – substantiv masculin

N. un	copil	nişte	copii	copilul	copiii
D. unui	copil	unor	copii	copilului	copiilor
G. unui	copil	unor	copii	copilului	copiilor
A. un	copil	nişte	copii	copilul	copiii

canton – substantiv neutru

N. un	canton	nişte	cantoane	cantonul	cantoanele
D. unui	canton	unor	cantoane	cantonului	cantoanelor
G. unui	canton	unor	cantoane	cantonului	cantoanelor
A. un	canton	nişte	cantoane	cantonul	cantoanele

auriu – adjectiv

N. (un)	auriu	(nişte)	aurii	auriul	auriii
D. (unui)	auriu	(unor)	aurii	auriului	auriilor
G. (unui)	auriu	(unor)	aurii	auriului	auriilor
A. (un)	auriu	(nişte)	aurii	auriul	auriii

N. (o)	aurie	(nişte)	aurii	auria	auriile
D. (unei)	aurii	(unor)	aurii	auriei	auriilor
G. (unei)	aurii	(unor)	aurii	auriei	auriilor
A. (o)	aurie	(nişte)	aurii	auria	auriile

negru – adjectiv

N. (un)	negru	(nişte)	negri	negrul	negrii
D. (unui)	negru	(unor)	negri	negrului	negrilor
G. (unui)	negru	(unor)	negri	negrului	negrilor
A. (un)	negru	(nişte)	negri	negrul	negrii

N. (o)	neagră	(nişte)	negre	neagra	negrele
D. (unei)	negre	(unor)	negre	negrei	negrelor
G. (unei)	negre	(unor)	negre	negrei	negrelor
A. (o)	neagră	(nişte)	negre	neagra	negrele

turna – verb

Prezent		*Perfectul simplu*	*Imperfect*	*M.m.c.p.*
eu	torn	turnai	turnam	turnasem
tu	torni	turnaşi	turnai	turnaseşi
el	toarnă	turnă	turna	turnase
noi	turnăm	turnarăm	turnam	turnaserăm
voi	turnaţi	turnarăţi	turnaţi	turnaserăţi
ei	toarnă	turnară	turnau	turnaseră

Conjunctiv	*Imperativ*
eu să torn	
tu să torni	toarnă
el să toarne	
noi să turnăm	
voi să turnați	turnați
ei să toarne	

Infinitiv	*Infinitiv lung*	*Participiu*	*Gerunziu*
a turna	turnare	turnat	turnând

merge – verb

Prezent	*Perfectul simplu*	*Imperfect*	*M.m.c.p.*
eu merg	mersei	mergeam	mersesem
tu mergi	merseși	mergeai	merseseși
el merge	merse	mergea	mersese
noi mergem	merserăm	mergeam	merseserăm
voi mergeți	merserăți	mergeați	merseserăți
ei merg	merseră	mergeau	merseseră

Conjunctiv	*Imperativ*
eu să merg	
tu să mergi	mergi
el să meargă	
noi să mergem	
voi să mergeți	mergeți
ei să meargă	

Infinitiv	*Infinitiv lung*	*Participiu*	*Gerunziu*
a merge	mergere	mers	mergând

vedea – verb

Prezent	*Perfectul simplu*	*Imperfect*	*M.m.c.p.*
eu văd	văzui	vedeam	văzusem
tu vezi	văzuşi	vedeai	văzuseşi
el vede	văzu	vedea	văzuse
noi vedem	văzurăm	vedeam	văzuserăm
voi vedeţi	văzurăţi	vedeaţi	văzuserăţi
ei văd	văzură	vedeau	văzuseră

Conjunctiv	*Imperativ*
eu să văd	
tu să vezi	vezi
el să vadă	
noi să vedem	
voi să vedeţi	vedeţi
ei să vadă	

Infinitiv	*Infinitiv lung*	*Participiu*	*Gerunziu*
a vedea	vedere	văzut	văzând

şti – verb

Prezent	*Perfectul simplu*	*Imperfect*	*M.m.c.p.*
eu ştiu	ştiui	ştiam	ştiusem
tu ştii	ştiuşi	ştiai	ştiuseşi
el ştie	ştiu	ştia	ştiuse
noi ştim	ştiurăm	ştiam	ştiuserăm
voi ştiţi	ştiurăţi	ştiaţi	ştiuserăţi
ei ştiu	ştiură	ştiau	ştiuseră

Conjunctiv	*Imperativ*
eu să ştiu	
tu să ştii	ştii
el să ştie	

noi să știm
voi să știți   știți
ei să știe

Infinitiv	*Infinitiv lung*	*Participiu*	*Gerunziu*
a ști	știre	știut	știind

SURSE BIBLIOGRAFICE

[1] DUMITRU IRIMIA, *Structura gramaticală a limbii române. Verbul,* Iași, 1976.

[2] ION COTEANU, *Gramatica de bază a limbii române,* București, 1982.

[3] V. GUȚU ROMALO, *Morfologia structurală a limbii române (substantiv, adjectiv, verb),* București, 1968.

[4] GR. MOISIL, *Probleme de traducere automată. Conjugarea verbelor în limba română,* în SCL, XI, 1960, nr. 1, p. 7-25.

[5] *Gramatica limbii române,* București, Editura Academiei, 1966.

[6] GR. MOISIL, *Problèmes posés par la traduction automatique. La declinaison en roumain écrit,* în CLTA, II, 1962.

[7] M. MANOLIU, *Morfemul în lingvistica modernă,* în LR, XII, 1963, nr. 1.

[8] *Dicționar ortografic, ortoepic și morfologic al limbii române,* București, Editura Academiei, 1982.

[9] V. BREBAN, *Dicționar general al limbii române,* București, 1992.

[10] *Dicționar ortografic cu elemente de ortoepie și morfologie.* Redacția principală a Enciclopediei Sovietice Moldovenești, Chișinău, 1991.

[11] J. D. ULLMAN, *Principles of Database Systems,* Computer Science Press, 1982.

[12] COLING BUDAPEST. *Proceedings of the 12th International Conference on Computational Linguistics,* 1988.

[13] J. S. BIEN, *Toward Computerizied Dictionaries for Inflexional Languages. Internal design philosophy,* în *Machine Translation,* Cambridge University Press, 1987.

[14] M. KAY, *The Dictionary of the Future and the Future of the Dictionary,* în *Machine Translation,* Cambridge University Press, 1987.

[15] A. ZAMPOLLI, *Lexicological and Lexicographical Activities at the Istituto di Linguistica,* în „Linguistica Computazionale”, 1983/3.

Universitatea „Babeș-Bolyai”
Facultatea de Litere
Centrul de Analiză a Textului
Cluj-Napoca, str. Horea, 31